

A Recommendation Approach in Social Learning Based on K-Means Clustering

Sonia Souabi*, Asmaâ Retbi, Mohammed Khalidi Idrissi, Samir Bennani

RIME TEAM-Networking, Modeling and e-Learning Team- MASI Laboratory- Engineering 3S Research center. Mohammadia School of Engineers (EMI), Mohammed V University, Rabat, 10000, Morocco

ARTICLE INFO

Article history:

Received: 01 December, 2020

Accepted: 14 January, 2021

Online: 05 February, 2021

Keywords:

Social Networks

Hybrid Recommendation System

K-means

Correlation

Co-occurrence

ABSTRACT

E-learning, among the most prominent modes of learning, offers learners the opportunity to attend online courses. To improve the quality of online learning, social learning through social networks promotes interaction and collaboration among learners. As part of the learning process management in these environments, the implementation of recommendation systems facilitates the provision of content adapted to the needs and requirements of learners and generates recommendations likely to arouse their interest. Many researchers have been involved in several recommendation techniques such as the development of Machine Learning algorithms and the incorporation of social interactions between learners. However, the behavior within a learning environment can diverge from one learner to another. This must therefore be taken into consideration when generating recommendations, i.e., it is initially important to form groups of homogeneous learners prior to proposing recommendations. In this respect, the recommendations generated will be more appropriate to the learners' profiles and level of interaction. On this basis, we raise an important issue which is the importance of grouping learners into homogeneous groups in a recommendation system. In the recommendation system we advocate, we group learners based on the degree of interaction within the learning environment before generating the recommendation list based on a hybrid approach for each cluster. The overall system is, therefore, based on the identification of communities based on the k-means algorithm and the generation of recommendations list for each community separately. Finally, we compare the results of the system integrating the classification of learners as a preliminary step to the system excluding the k-means algorithm. The results reveal that the integration of the clustering algorithm leads to improvements in terms of performance and accuracy.

1. Introduction

This paper is an extension of the work originally presented at the Fourth International Conference on Intelligent Systems and Computer Vision [1]. Currently, e-learning is emerging on a large scale around the world. When face-to-face learning is not available, e-learning allows students to bring order to their ideas and to continue to lead students in their learning process [2], [3]. Social networks, a modernized version of learning, fully promotes interaction and collaboration among learners [4], [5]. When a traditional learning platform is sometimes unable to provide effective collaboration between learners, social networks offer a multitude of options for learners encouraging responsiveness between them. However, the availability of online courses is not the only factor that contributes to the success of the learning

process. Learners require a more highly organized learning environment responsive to their needs and profile. In this respect, recommendation systems are the perfect medium to perform this task and to manage learning resources within e-learning [6], [7]. The role of recommendation systems is to filter the information that aims to present different learning objects. Through recommendation systems, a certain organization is implemented within the learning environment and learners are more able to explore what is most interesting to them. Many recommendation systems have been proposed in the literature, including: the content-based approach, collaborative filtering and hybrid approaches [8]-[10]. Within the e-learning context, each proposed recommendation system addresses specific questions and handles an underlying issue identified at the recommendation system level. By way of example, several researchers address collaborative filtering to generate recommendations to learners [11]. Others

*Corresponding Author: Sonia Souabi, soniasouabi@research.emi.ac.ma

focus their attention on hybrid approaches to improve the performance of their recommendation system. Machine Learning algorithms have also been proposed as part of learning recommendation systems, including supervised and unsupervised algorithms [12]. Most recommendation systems use explicit feedback from learners, i.e., assessment by learners to generate recommendations. However, explicit feedback is sometimes an unreliable indicator for recommendations. On the other hand, an e-learning environment can include hundreds or thousands of learners with varying degrees of interactivity, constituting divergent groups, interactive and non-interactive learners. Most studies conducted on recommendation systems do not consider that point with respect to online learning. In this regard, we propose to group learners into homogeneous groups before generating recommendations. Our work aims at implementing a preliminary step in the calculation of recommendations, namely the detection of communities with the same level of interactivity. In our previous work, we proposed a hybrid recommendation system based on the activities performed by the learners. Within the same framework, the current work presents a continuity of the previous work while improving the old system with a new recommendation strategy integrating the k-means algorithm as a preliminary step. In this sense, we propose to classify learners according to well-defined criteria of interactivity within a social learning network, and then generate recommendations for each community separately. In this way, each community will receive its own recommendations adapted to its specific needs. The approach we propose brings considerable added value as it categorizes learners according to their level of interactivity before calculating recommendations.

The document is divided as follows: Section 2 deals with the research work related to the recommendation systems proposed in e-learning, especially those that consider Machine Learning algorithms. Section 3 explains the global recommendation approach based on correlation, co-occurrence and the k-means algorithm, section 4 presents the results of the tests performed on the approach without k-means and the approach with k-means with interpretation, and section 5 summarizes the work done in the paper.

2. Literature review

2.1. Recommender systems in E-learning

There are several types of recommender systems: content-based recommender systems, collaborative filtering recommender systems and hybrid recommender systems. Hybrid recommender systems generally combines between content-based techniques and collaborative filtering approaches. In E-learning, all these types of recommendations were addressed, but differently from one proposition to another. Some researchers discuss the analysis of activities in collaborative filtering within the educational field [13]. This work aims to use ontology and the semantic web to provide efficient recommendations. Others propose a recommendation approach to guide learners in developed countries to select more appropriate resources [14]. Calculations are based on developed knowledge and rating predictions. A recommendation system was suggested based mainly on two primary steps: pre-processing and prediction [15]. Multiple algorithms was used: SVM, KNN, Random Forest, and Naïve Bayes. A personalized recommendation system was developed

based mainly on ontology using the java programming language [16]. Learning style and level of knowledge were addressed in a proposed recommender system [17]. The approach contains four modules relating to courses and learners.

2.2. Related works

In Machine Learning, unsupervised learning includes algorithms that should perform based on unannotated examples. K-means, one of the important unsupervised algorithms, is used to identify clusters with the similar characteristics and properties by working with gravity centres. The optimal number of clusters is usually obtained through several methods, including: elbow method, average silhouette method, gap statistic method. In online learning, k-means is handled for many purposes, for instance classifying learners according to their attitudes, their performance, interaction rates (...). In the literature, there is a multitude of works addressing k-means as an algorithm for classifying not only learners, but also items, users, in different contexts and areas, such as recommender systems.

A learning content recommendation system was implemented within a learning platform to generate recommendations in an intelligent way based on the learners' interest [18]. The proposed system aims to combine ontology and clustering algorithms using the collected ontology. A new recommendation system that combines the content-based approach and the k-means algorithm simultaneously was proposed [19]. The approach consists of performing a transformation on user data. Then, learners are grouped into clusters based on the content approach and the clustering algorithm, and thus recommendations will be generated from the detected clusters. Some researchers focus on methods to optimize k in the clustering algorithms in order to maintain the variance of each cluster [20]. The clustering was based on movie genres and tags. The objective is to correctly evaluate the recommendation algorithm based on user classification and use sophisticated measures such as mean square error, proximity centrality. Abnormal profiles in the context of recommendation systems was detected. A hybrid recommendation system that integrates both the k-means algorithm and ant colony optimization was suggested [21]. The evaluation was based on several measures including precision, recall and accuracy. Several approaches are based on clustering techniques in recommendation systems that are mainly based on group preferences. Several algorithms were compared to identify which algorithm gives better accuracy and higher results [22]. Some use k-means, others use k-NN, and the results were compared in terms of performance and number of clusters obtained. Instead of working with a traditional clustering algorithm, a multi-clustering approach working on a set of clusters was proposed [23]. The advantage of this algorithm is its ability to visualize the neighbourhood in a clearer and more refined way, and the time efficiency is high. Contextual information was used to generate more relevant recommendations [24]. The proposed technique seeks to use the k-means algorithm to cluster contexts and generate new user matrices.

Based on researches in terms on recommender systems based on clustering algorithms, it comes out that k-means has been approached in many areas, such as e-learning, e-commerce, movies in order to generate more appropriate recommendations. Thus, we can notice that clustering algorithms has been adopted in recommender systems for different purposes in all the cited works.

On the other hand, clustering algorithms have not been adequately addressed in e-learning. In our context, we propose to use the k-means algorithm within a pure learning context and to combine it with a hybrid recommendation approach in order to generate more relevant recommendations. Our goal is to classify learners according to their degree of interactivity before calculating recommendations to generate a more relevant list of recommendations.

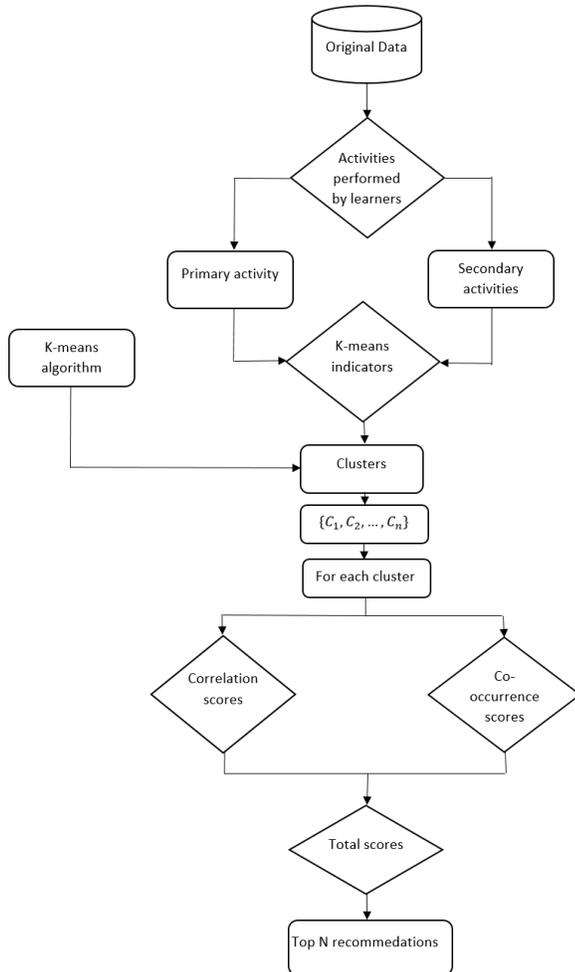


Figure 1: The recommendation system model proposed

3. Our proposed approach

3.1. K-means algorithm

Before proceeding to the description of our recommendation system, we will first define the k-means algorithm. The k-means algorithm is considered among the most widespread clustering algorithms. It consists of the analysis of a dataset containing several descriptors. The final goal is to group similar data in the same clusters. The idea is straightforward, all you need to do is to calculate the distance between the descriptors of two data to decide whether they are part of the same cluster. In a set of data, and to implement the k-means algorithm, there are the so-called centroids. Centroids are the key elements of the algorithm. The k-means relies on the centroids to detect the data closest to each centroid, and this loop continues until there is no further improvement in the distances between similar data, i.e. until the

algorithm converges. The algorithm below summarizes the different parts of the k-means algorithm (algorithm 1).

Algorithm 1: Concept of K-means Algorithm

Input

K optimum number of clusters

Output

Clusters or groups

Method

Selecting randomly K points. These are the centroids.

REPEAT

(1): Assign each point to the closest group according to centroids.

(2): Modifying centroids according to new clusters.

UNTIL CONVERGENCE

3.2. The proposed recommender system

The following flowchart (Figure 1) summarizes our recommendation approach.

The flowchart consists of several parts core to the proposed approach:

- The learners' actions in the database (sharing, communicating, downloading ...).
- The reconsideration of the initial data.
- The k-means algorithm.
- The different learning resources.
- The recommendation part.

In this part, we will describe each phase of the proposed approach separately:

- a- First phase: The first phase consists in detecting the learners' actions in the database that have been carried out within the learning environment, for instance: sharing learning resources, sending private messages to other learners, downloading resources, exploiting resources. These actions are then grouped into two main categories: primary and secondary actions. The primary action is the action that can directly reflect learners' real preferences, for example, liking content, searching for content. It is through this indicator that we can ascertain learners' preferences. Secondary actions are also indicators of learner preferences, but are less consistent than the primary action.
- b- Reconsideration of the initial data: This part consists in converging the data table into data that translate the interaction between learners and each learning object with respect to the selected actions, i.e. it is mandatory to restructure the database so that it meets our requirements for calculating recommendations.
- c- The k-means algorithm: After restructuring the database, the next step consists in grouping the learners in clusters by selecting a number of descriptors. Each learner has a number of descriptors which are used to build the most similar clusters. Therefore, the choice of descriptors is an obvious prerequisite for applying the k-means algorithm.
- d- Recommendations part: the last part consists of calculating the recommendations after detecting the clusters of the most similar learners. We will therefore generate the recommendations for each cluster separately. The idea is to consider the notions of correlation and co-occurrence in the calculation of recommendations, and thus to generate the final recommendations.

The hybrid recommendation algorithm combined with k-means algorithm is described below (algorithm 2).

Algorithm 2: Recommendation algorithm with k-means

Input

Learners data
Actions performed by learners

Output

Recommendations generated for each group

Method

- 1: Describing each learner by descriptors.
- 2: implementing the K-means algorithm based on previous list of learners and their descriptors.
- 3: Assign each learner to a specific cluster.

For each cluster

- 1: Calculate correlation and co-occurrence matrices.
- 2: Calculate recommendation scores based on previous step.
- 3: Find top N recommendations for each learner.

End for each

3.3. First stage: the k-means algorithm

➤ Define the descriptors

The database contains the learners' history, including the actions that are performed by the learners. We restructure the database to group learners according to two main descriptors:

- The first descriptor describes the level of interaction based mainly on the number of times an action has been performed.
- The second descriptor reflects the discrepancy between the actions performed by a learner.

Based on these two descriptors, it is possible to group learners into several clusters using the k-means algorithm within the learning platform. The indicators or descriptors are calculated as follows:

For a particular learner, the history of actions $\{a_1, a_2\}$ such as a_1 presents the primary action et a_2 is the secondary action, and the following resources $\{i_1, i_2\}$ (table 1 and table 2).

Table 1: Activities vs learning items structure

	a_1	a_2
i_1	x_1	x_2
i_2	x_3	x_4

Table 2: K-means descriptors

Learner 1	
First descriptor	$x_1 + x_3$
Second descriptor	$(x_1 + x_3) - (x_2 + x_4)/2$

For each learner, we will perform this operation, which allows us to classify learners according to these two indicators.

➤ Optimum number of K:

In order to properly apply the k-means algorithm, it is of paramount importance to define the optimal number k of clusters to be identified. One of the most successful methods for calculating the optimal number of k is the elbow method. It actually allows to specify the most adequate number of k to be considered since the

parameter k can change and can take several values. The elbow method consists in describing the variation according to the number of clusters. But in this case, what would be the optimal number? The optimal number k is simply the value from which the curve starts to take a constant value. So we will ensure that the value of k we selected represents the most accurate value for our context.

3.4. Second stage: The hybrid recommendation part

Now that the k-means algorithm is applied and clusters are detected, we now have the different groups of learners with similar behaviour in terms of interaction level. The rest of the work consists in generating recommendations through the calculation of recommendation scores for each learner pertaining to a given class. The proposed recommendation system is based on the calculation of correlation and co-occurrence scores through the analysis of the actions performed by the learners. In the following section, we will describe the mathematical model of our recommendation approach in detail:

The history matrix of learners =



	i_1	...	i_n
l_1	a_1 ... a_m	a_1 ... a_m	a_1 ... a_m
\vdots	\vdots	\vdots	\vdots
l_p	a_1 ... a_m	a_1 ... a_m	a_1 ... a_m

$\{i_1, \dots, i_n\}$: Learner objets.

$\{a_1, \dots, a_m\}$: History actions performed by learners.

$\{l_1, \dots, l_p\}$: Learners.

Referring to the basic matrix, which presents the learners' interaction with contents according to the selected actions, we generate recommendations for each cluster detected separately through the calculation of recommendation scores.

➤ Correlation

Correlation allows to identify the existing connection between two variables. Our context consists in calculating the correlation between the primary action and the secondary actions. The purpose is to situate the importance of the secondary actions in relation to the primary action to be used in the calculation of recommendations. We opted for the spearman correlation since it is the type of correlation that best suits our case, since it provides a way to evaluate the correlation in case of a non-normal distribution. We explain the notion of correlation in what follows (table 3 and eq. 1).

Table 3: General structure of correlation matrix

	a_1	...	a_m
a_1	1	...	$cor(a_1, a_m)$

⋮	⋮	⋮	
a_m	$cor(a_1, a_m)$...	1

$$Correlation\ score = Total\ history\ matrix \times \begin{bmatrix} 1 \\ \vdots \\ cor(a_1, a_m) \end{bmatrix} = \begin{bmatrix} s_{11} & \dots & s_{1n} \\ \vdots & \dots & \vdots \\ s_{l1} & \dots & s_{ln} \end{bmatrix} \quad (1)$$

$\{(s_{11}, \dots, s_{1n}), \dots, (s_{l1}, \dots, s_{ln})\}$ are scores of correlation for each particular learner towards each learning resource.

➤ Co-occurrence

Co-occurrence is defined as a measure to determine how many times two actions occur simultaneously in the same database. Learners generally do not react to the same actions in the same manner. They can be active in one action without engaging in another action. In this case, the co-occurrence between these two actions might be very weak. The role of co-occurrence is to define where two actions are linked at this level (table 4 and eq. 2).

Table 4: General structure of co-occurrence matrix

	a_1	...	a_m
a_1	occurrence (a_1)	...	co - occ(a_1, a_m)
⋮	⋮	⋮	
a_m	co - occ(a_1, a_m)	...	occurrence (a_m)

$$Co - occurrence\ score = Total\ binary\ history\ matrix \times \begin{bmatrix} occurrence(a_1) \\ \vdots \\ co - occ(a_1, a_m) \end{bmatrix} = \begin{bmatrix} p_{11} & \dots & p_{1n} \\ \vdots & \dots & \vdots \\ p_{l1} & \dots & p_{ln} \end{bmatrix} \quad (2)$$

To generate the total recommendation scores, we proceed as follows (eq. 3).

$$Total\ score = Correlation\ score + co - occurrence\ score = \begin{bmatrix} s_{11} + p_{11} & \dots & s_{1n} + p_{1n} \\ \vdots & \dots & \vdots \\ s_{l1} + p_{l1} & \dots & s_{ln} + p_{ln} \end{bmatrix} \quad (3)$$

From the scores obtained, we extract the top N recommendations according to scores ranking.

4. Tests and results

In this work, we propose a recommendation system that aims to integrate the k-means algorithm within the system to generate more pertinent and reliable recommendations. Our intention is to prove that the k-means was efficient in generating more relevant recommendations and results. In order to test our recommendation approach for both cases, without k-means and with k-means, we selected two groups from a social network containing data on learners' interaction with the learning environment:

- The very first database consists of a set of 100 learners with many learning objects. We limited the number of learning objects to 10 and two actions, the primary action being the

number of likes and the secondary action being the number of shares.

- The second database contains the same 100 learners from the first database, but interacting in a different social learning group. We selected some learning objects for the aforementioned actions: the number of likes and the number of shares.

The purpose of choosing these two databases is to highlight the performance of our recommendation system which incorporates the k-means algorithm, and which emphasizes the importance of grouping learners with similar attitudes in the same cluster. Homogeneity will be a major advantage in this case in terms of generated recommendations.

➤ Recommendation system evaluation:

The evaluation of the recommendation system is a prerequisite to measure its performance. In this respect, we divide the database into two major parts:

- The first part which consists in creating the recommendation model by referring to 80% of the data.
- The second part consists in measuring the accuracy and performance of the system by referring to 20% of the data.

We highlight the following measures (eq. 4 and eq. 5) according to specific parameters (Table 5).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

Table 5: Accuracy and precision parameters

	Recommended	Not recommended
Preferred	True positive (TP)	False positive (FP)
Not preferred	False positive (FP)	True positive (TP)

Before moving on to the application of k-means, it is important to identify the optimal number of clusters. In this sense, we use the elbow method which allows to plot the variation as a function of the K number. The optimal value we reach for both databases through the elbow method is 2.

In what follows, we will compare the results obtained through the application of the recommendation system not considering the k-means and the recommendation system identifying the number of clusters in the database before generating the recommendations (Table 6).

Table 6: Accuracy and precision for both recommender systems in the two databases

Database		Accuracy	Precision
First database	Without clustering	0,375	0,5
	Cluster 1	0,4	0,52
	Cluster 2	0,416	0,526
	Average between	0,408	0,523

	cluster 1 and cluster 2		
Second database	Without clustering	0,3	0,44
	Cluster 1	0,40625	0,55
	Cluster 2	0,406	0,525
	Average between cluster 1 and cluster 2	0,406	0,5375

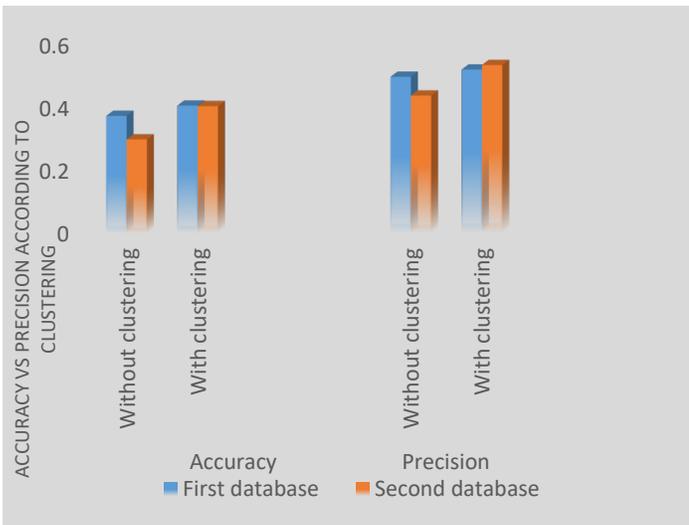


Figure 2: Accuracy and precision according to the two types of recommender systems (with k-means and without k-means)

Based on the results obtained (Figure 2), it appears that the recommendation system based on the classification of learners provides a higher performance compared to the recommendation system where community detection is not considered. A difference of $0.408-0.375=0.033$ in terms of accuracy and $0.526-0.5=0.026$ in terms of precision. Apparently, the difference seems insignificant since the database is not of considerable size. Therefore, if the test is performed on a larger database the difference will be more noticeable. The same holds true for the second database where the difference in accuracy is $0.406-0.3=0.106$ and the difference in precision is $0.5375-0.44=0.0975$. These results highlight the importance of grouping learners according to their degree of interactivity before proceeding with the assignment of recommendations. Grouping learners with similar patterns of behaviour within the same community allows the generation of more adapted recommendations to their needs. Generating recommendations to learners with no prior classification reduces the performance of the recommendation system and provides recommendations that do not necessarily fit the needs of each community separately from the other. We recognize that the k-means algorithm brings considerable added value to the recommendation system due to the accuracy and performance improvements achieved. Grouping learners in homogeneous clusters facilitates the generation of recommendations.

5. Conclusion

The following work proposes a hybrid recommendation approach merging learner classification as a preliminary step before proceeding with recommendation calculations. In addition to the implicit feedback from learners that was integrated into the initial recommendation system, our approach proposes to group learners into communities with the corresponding degree of interactivity to improve the quality of the recommendations generated. Our approach consists in classifying learners according to specific interactivity criteria, then calculating recommendations for each community based on all the activities performed by the learners within the learning social network. On the one hand, the two notions of correlation and co-occurrence are combined in a unique hybrid recommendation system valuing the implicit feedback from the learners. On the other hand, the approach fosters the detection of homogeneous communities and clusters in order to generate recommendations more adapted to each community separately. The test was carried out on two databases of two different social learning groups, each consisting of 100 learners. The objective was to collect implicit feedback from learners within the learning environment. The results indicated that the recommendation approach based on the k-means algorithm leads to higher performance than the system without the k-means algorithm. This highlights the importance of classifying learners into homogeneous groups before generating recommendations. Our future work will consist of:

- Testing our recommendation approach on a larger database.
- Applying other algorithms other than k-means.

References

- [1] S. Souabi, A. Retbi, M.K. Idrissi, S. Bennani, "A Recommendation Approach in Social Learning Based on K-Means Clustering," in 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), IEEE, Fez, Morocco: 1–5, 2020, doi:10.1109/ISCV49265.2020.9204203.
- [2] D.K. Abdul-Rahman Al-Malah, S. Ibrahim Hamed, H.Th.S. Alrikabi, "The Interactive Role Using the Mozabook Digital Education Application and its Effect on Enhancing the Performance of eLearning," International Journal of Emerging Technologies in Learning (IJET), 15(20), 21, 2020, doi:10.3991/ijet.v15i20.17101.
- [3] E. Edelhauser, L. Lupu-Dima, "Is Romania Prepared for eLearning during the COVID-19 Pandemic?," Sustainability, 12(13), 5438, 2020, doi:10.3390/su12135438.
- [4] A. Muradi, H. Hasanzada, "The Role of Social Networks in Learning English Language," 7(1), 10, 2020.
- [5] C.G. Brinton, M. Chiang, "Social learning networks: A brief survey," in 2014 48th Annual Conference on Information Sciences and Systems (CISS), IEEE, Princeton, NJ, USA: 1–6, 2014, doi:10.1109/CISS.2014.6814139.
- [6] A. Sekhavatian, M. Mahdavi, "APPLICATION OF RECOMMENDER SYSTEMS ON E-LEARNING ENVIRONMENT," 10, 2011.
- [7] M.H. Ansari, M. Moradi, O. NikRah, K.M. Kambakhsh, "CodERS: A hybrid recommender system for an E-learning system," in 2016 2nd International Conference of Signal Processing and Intelligent Systems (ICSPIS), IEEE, Tehran, Iran: 1–5, 2016, doi:10.1109/ICSPIS.2016.7869884.
- [8] D.M. Kandakatla, K. Bandi, "A Content Based Filtering and Negative Rating Recommender System for E-learning Management System," 7, 2018.
- [9] S.K. Gorripati, V.K. Vatsavayi, "Community-Based Collaborative Filtering to Alleviate the Cold-Start and Sparsity Problems," 12(15), 9, 2017.
- [10] A.N. Ngaffo, W.E. Ayeb, Z. Choukair, "A Bayesian Inference Based Hybrid Recommender System," IEEE Access, 8, 101682–101701, 2020, doi:10.1109/ACCESS.2020.2998824.
- [11] S. Sharma, A. Mahajan, "A Collaborative Filtering Recommender System for Github," 5(8), 7, 2017.

- [12] S.S. Khanal, P.W.C. Prasad, A. Alsadoon, A. Maag, "A systematic review: machine learning based recommendation systems for e-learning," *Education and Information Technologies*, 2019, doi:10.1007/s10639-019-10063-9.
- [13] M. Brik, M. Touahria, "Contextual Information Retrieval within Recommender System: Case Study 'E-learning System,'" *TEM Journal*, 1150-1162, 2020, doi:10.18421/TEM93-41.
- [14] J.-P. Niyigena, Q. Jiang, *A Hybrid Model for E-Learning Resources Recommendations in the Developing Countries*, Association for Computing Machinery, New York, NY, USA: 21-25, 2020.
- [15] K. Chaudhary, N. Gupta, "E-Learning Recommender System for Learners: A Machine Learning based Approach," *International Journal of Mathematical, Engineering and Management Sciences*, 4(4), 957-967, 2019, doi:10.33889/IJMEMS.2019.4.4-076.
- [16] O.C. Agbonifo, M. Akinsete, "Development of an Ontology-Based Personalised E- Learning Recommender System," **38**(1), 12, 2020.
- [17] Sunil, M.N. Doja, "An Improved Recommender System for E-Learning Environments to Enhance Learning Capabilities of Learners," in: Singh, P. K., Panigrahi, B. K., Suryadevara, N. K., Sharma, S. K., and Singh, A. P., eds., in *Proceedings of ICETIT 2019*, Springer International Publishing, Cham: 604-612, 2020.
- [18] Saintgits College of Engineering, India, A. S Nath, "A Personalized Course Recommender System for E-Learning," *International Journal of Networks and Systems*, 11-14, 2019, doi:10.30534/ijns/2019/03832019.
- [19] P. Cao, D. Chang, "A Novel Course Recommendation Model Fusing Content-Based Recommendation and K-Means Clustering for Wisdom Education," in: Zhang, J., Dresner, M., Zhang, R., Hua, G., and Shang, X., eds., in *LISS2019*, Springer Singapore, Singapore: 789-809, 2020.
- [20] D. Cintia Ganesha Putri, J.-S. Leu, P. Seda, "Design of an Unsupervised Machine Learning-Based Movie Recommender System," *Symmetry*, **12**(2), 185, 2020, doi:10.3390/sym12020185.
- [21] M.S. Kumar, J. Prabhu, "A hybrid model collaborative movie recommendation system using K-means clustering with ant colony optimisation," 18, 2020.
- [22] Y. Yang, D. Hooshyar, J. Jo, H. Lim, "A group preference-based item similarity model: comparison of clustering techniques in ambient and context-aware recommender systems," *Journal of Ambient Intelligence and Humanized Computing*, **11**(4), 1441-1449, 2020, doi:10.1007/s12652-018-1039-1.
- [23] U. Kuzelewska, "Dynamic Neighbourhood Identification Based on Multi-clustering in Collaborative Filtering Recommender Systems," in: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., and Kacprzyk, J., eds., in *Theory and Applications of Dependable Computer Systems*, Springer International Publishing, Cham: 410-419, 2020.
- [24] E. Kannout, "Context Clustering-based Recommender Systems," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, 85-91, 2020, doi:10.15439/2020F54.